

# **WAVE FILE PLAYER**

## **MED-STATE NOTATION™ PROCEDURE**

**SOF-732-2 Manual**  
**DOC-029**  
**Rev. 2.3**

Copyright © 2010  
All Rights Reserved

MED Associates, Inc.  
P.O. Box 319  
St. Albans, Vermont 05478  
[www.med-associates.com](http://www.med-associates.com)

Trademarks: MedState Notation™, MED Associates, Inc.  
Registered Trademark:  
MED-PC®, MED Associates, Inc.  
Delphi®, Borland International, Inc.



## TABLE OF CONTENTS

<b>Chapter 1 .....</b>	<b>1</b>
Introduction.....	1
Overview of the Procedure .....	2
<b>Chapter 2 .....</b>	<b>3</b>
Getting Started.....	3
Software Installation .....	3
Backing Up the Software .....	3
<b>Chapter 3 .....</b>	<b>4</b>
Beginning & Running an Experiment .....	4
Creating a List File .....	4
Translating the MED-PC (.mpc) File .....	5
Using the MED-PC Load Wizard .....	6
Playing Wave Files .....	10
<b>Chapter 4 .....</b>	<b>11</b>
Function Reference .....	11
SetWAVFile .....	11
IsReady .....	11
PlayWave.....	12
<b>Chapter 5 .....</b>	<b>13</b>
MEDPCSOUNDSAMPLE.MPC .....	13



## CHAPTER 1

### *Introduction*

The purpose of this manual is to give an explanation of the MED State Notation™ Procedures that comprise the SOF-732-2 Wave File Player. The files in this package can be found on the disk provided by MED Associates, Inc.

These procedures are intended to be run in MED Associates MED-PC® software. The latest version of MED-PC® gives researchers the ability to use pre-programmed procedures such as these to make hardware control and data collection easy. These pre-programmed procedures can also be modified to meet the evolving demands of a research protocol. Again, it is the intent of this manual to explain exactly what these procedures implement, and provide guidance into how to interpret what the program code achieves in order to let the user determine how to modify them to match their research protocol demands. The manual provides some examples of editing and modifying the procedure's programming code. The manual also defines the elements in the raw data file produced by these procedures.

In addition to this manual, refer to the **MED-PC® User's Manual** for the installation of the MED-Associates interface drivers, the MED-PC Software, and the Delphi® Compiler. Also refer to the User's Manual for instructions on developing a Hardware Configuration. Data file structure, file-saving format, and other related options are also determined by the Hardware Configuration. Running the Hardware Configuration software utility that accompanies MED-PC sets the Hardware Configuration. Its purpose is to assign the inputs and outputs on the interface cards in the interface cabinet for each task controlled by MED-PC. The particular type of interface card that is supplied in the interface cabinet may vary; please refer to the User's Manual provided for instructions on how to configure the type of card that is in the cabinet. A valid Hardware Configuration must exist in order for MED-PC to interface correctly with the MED Associates, Inc. hardware. This means that one should take the time to create a valid Hardware Configuration before attempting to run the procedures included in this package.

Should there be any problems, the staff at MED Associates, Inc. is available to answer any questions that may arise. Please e-mail us at [support@med-associates.com](mailto:support@med-associates.com) with a detailed description of the problem or desired goals so that concise and detailed information may be provided.

The Wave File Player procedure is designed to be as easy to use as possible. MED Associates, Inc. understands that researchers do not have the time to devote to programming and hardware design, and for that reason, we have undertaken that burden for you. We sincerely hope that you are satisfied with the products and services we provide, and look forward to meeting your future experimental needs as your research program evolves.

**Overview of the Procedure**

Wave File Player is a utility for playing .WAV format sound files from MED-PC.

The WAV File Player allows for positive or negative reinforcement with sound through the PC's sound system (sound card). The included program has six sounds. The sounds are played sequentially from the list file until a K1 pulse is issued.

The program allows the user to play any WAV file, not just the ones provided with the program. Furthermore, the program can be modified to use sound as a stimulus or reinforcer in many other ways. For example, the user could set up a protocol to play a WAV file when the animal enters a specific side of a shuttle box.

## CHAPTER 2

### *Getting Started*

#### Software Installation

Please refer to the **MED-PC User's Manual** for a complete guide to installing the MED-PC software, building a valid Hardware configuration with the Hardware Configuration utility, and opening and compiling a MSN procedure in the Trans-IV utility.

To install the Wave File Player Procedure, insert the CD into the CD-ROM drive and click **Install the Wave File Player Software**. The installer will search for the MED-PC installation directory, and install the following files:

*Figure 2.1 - MEDPC Sound Files Installed*

FILE	Description	LOCATION
MEDPCSound.dll	Library file, exports 3 functions: SetWAVFile, IsReady, PlayWave	MED-PC installation directory
MEDPCSound_Iface.hed	Header file, defines interface to 3 exported functions	MED-PC installation directory
MEDPCSoundSample.mpc	MPC code, sample described below in this document	MED-PC \MPC directory
WavFiles.txt	Wav List file, used by MEDPCSoundSample.mpc	MED-PC installation directory
chimes.wav, chord.wav, ding.wav, ringin.wav, ringout.wav, tada.wav	.WAV files, used by MEDPCSoundSample.mpc	MED-PC installation directory

The installer will also add the following line to the MED-PC \User.pas file:

*Figure 2.2 - Line Added to User.pas File*

Inserted reference	User.pas section
{ \$I MEDPCSOUND_IFACE.HED }	{ Place \$I Filename.HED files here. }

#### Backing Up the Software

Med Associates strongly encourages creating backup copies of the programs in case of disk failure. Having copies of the original programs may be useful in the future should modifications be made to the existing programs.

## CHAPTER 3

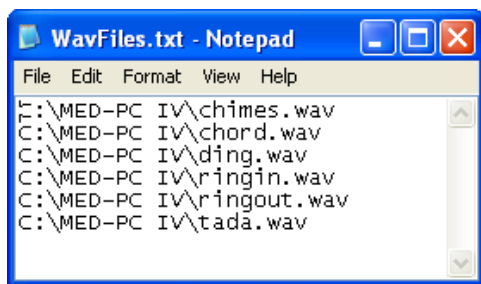
### ***Beginning & Running an Experiment***

#### **Creating a List File**

The sample list included with the Wave File Player (WavFiles.txt) may be used; however custom sound files can be created. To do this, begin by creating a text file that lists the wave (.wav) file names. The text file can be created using Trans IV, Notepad, etc. Place one wave file name per line, including the drive letter and full file path. For example, the WavFiles.txt file installed to the MED-PC installation directory is shown in Figure 3.1.

*Figure 3.1 - Example WAV List File*

---



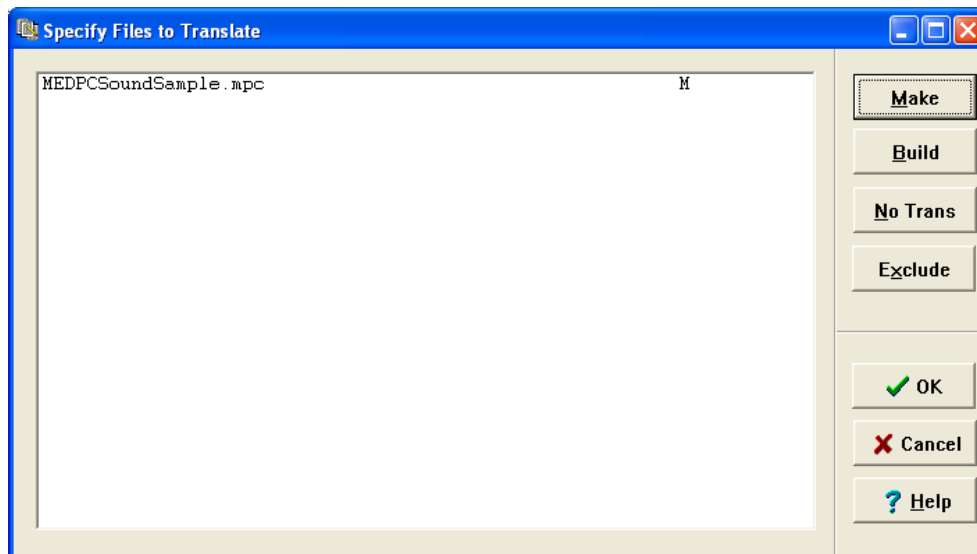


## Translating the MED-PC (.mpc) File

Programs written in MedState Notation must be translated using Trans IV before they can be executed in this application. Open Trans IV icon and select **Translation | Translate and Compile**.

Select the program(s) to use for the experiment and click **M**ake. Click **OK** to start the translator, and it will automatically parse the MedState Notation and then open to a DOS screen to compile the Pascal code. Depending on the speed of the computer, each of these steps may not be seen. If any problems are encountered during this process, refer to the on-screen help menu or the **MED-PC Version User's Manual**, or contact MED Associates, Inc. for assistance.

*Figure 3.2 - Trans IV Control Panel for Translating and Compiling MedState Notation Code*



## Using the MED-PC Load Wizard

MED-PC is designed to help the researcher run an experiment by guiding selection choices through its Experiment Loading Wizard. This section will describe how to initiate the MEDPCSOUNDSAMPLE.mpc application, however the following steps that will also apply to all other .mpc procedures.

Open MED-PC and the MED-PC Experiment Loading Wizard's Welcome screen, shown in Figure 3.3 will appear.

*Figure 3.3 - The MED-PC Loading Wizard Welcome Screen*



To avoid this load wizard, deselect the checkbox labeled **Run this experiment automatically when starting MED-PC**. Close this screen by clicking the **Close** button. Closing this screen immediately reveals the MED-PC Run-Time Screen shown in Figure 3.9. If the choice to continue with the Loading Wizard is made, then click the **Next** button.

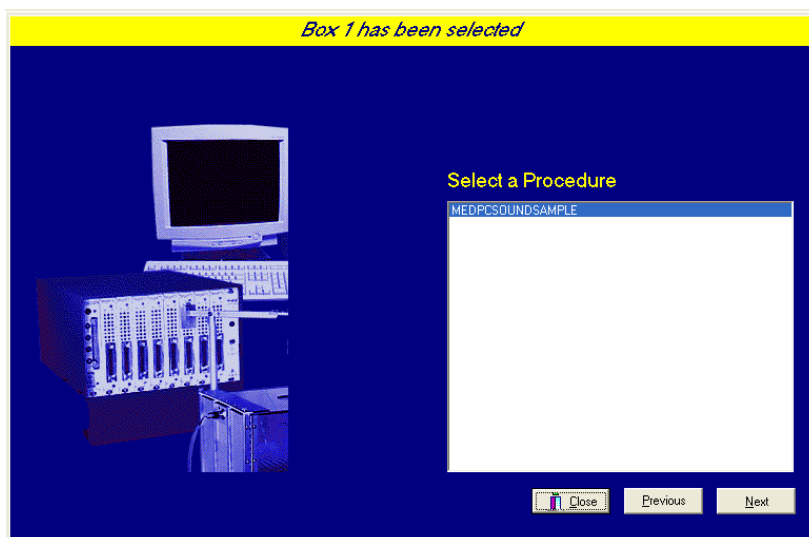
The Box Selection screen will appear next, as shown in Figure 3.4. From this screen the researcher chooses which boxes will be used in the experiment. Select the boxes that will run the experiment by clicking in the radio button next to the box number. The figure shows that the Hardware Configuration included only 1 box, which was selected. Click **Next** to continue.

*Figure 3.4 - The Box Selection Screen*



The Select a Procedure screen appears next, as seen in Figure 3.5. This is where the application to be run is selected. The screen displays a list of all the currently compiled procedures. Select the desired procedure and then click **Next**.

*Figure 3.5 - The Select a Procedure Screen*



The Enter Experiment Data Screen should display next, as shown in Figure 3.6. The purpose of this screen is to allow annotations to be added to the data file that is produced by MED-PC. These annotations will help identify the Subject, Experiment, and Experiment Group upon which data was collected. Comments can be added here as well, and the data file can be given a customized file name to help identify it from other data files. Enter the information desired, and click **Next**.

Figure 3.6 - Enter Experiment Data Screen



Box 1 and MEDPCSOUNDSAMPLE have been selected

Subject: Subject\_1

Experiment: Experiment\_1

Group: Group\_1

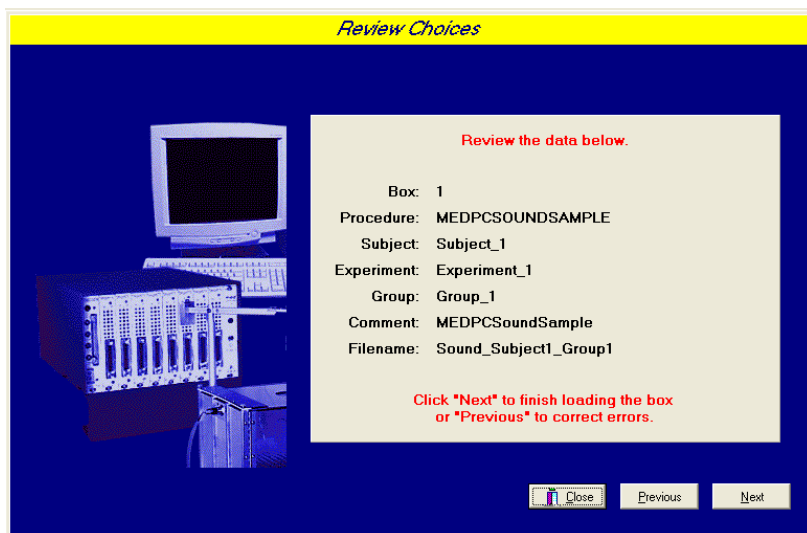
Comments: MEDPCSoundSample

Optional Custom Filename: Sound\_Subject1\_Group1

Close Previous Next

The next screen to appear is the Review Choices screen, as seen in Figure 3.7. This is a method of confirming that the information received from the Box/Procedure Selected is correct. If it is not correct, select **Previous**, and edit the data. If it is correct, select **Next**.

Figure 3.7 - Review Choices Screen



Review Choices

Review the data below.

Box: 1

Procedure: MEDPCSOUNDSAMPLE

Subject: Subject\_1

Experiment: Experiment\_1

Group: Group\_1

Comment: MEDPCSoundSample

Filename: Sound\_Subject1\_Group1

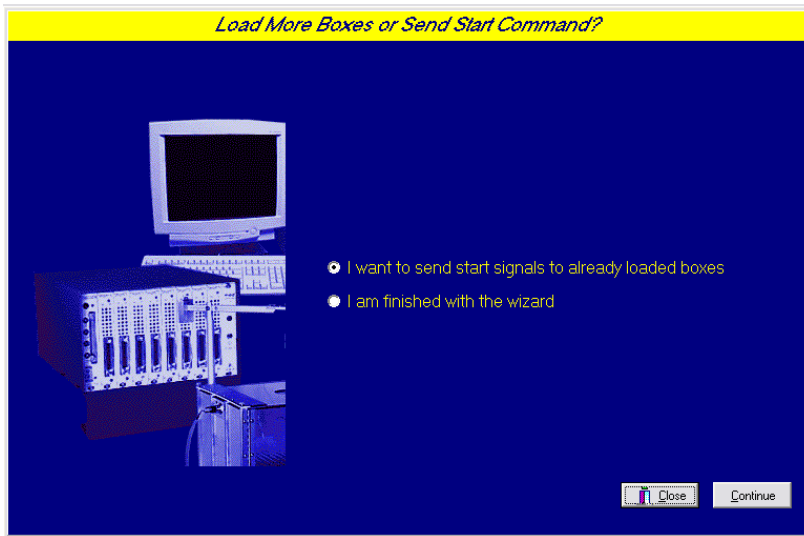
Click "Next" to finish loading the box  
or "Previous" to correct errors.

Close Previous Next

The Send Start Command Screen appears next. The options available on the screen vary depending upon how many boxes are described in the Hardware Configuration.

In this example only one box is described in the Hardware Configuration, so Figure 3.8 will appear next. If more than one box is in the Hardware Configuration, then Figure 3.9 will appear.

*Figure 3.8 - Send Start Command Screen for Single Box Configuration*

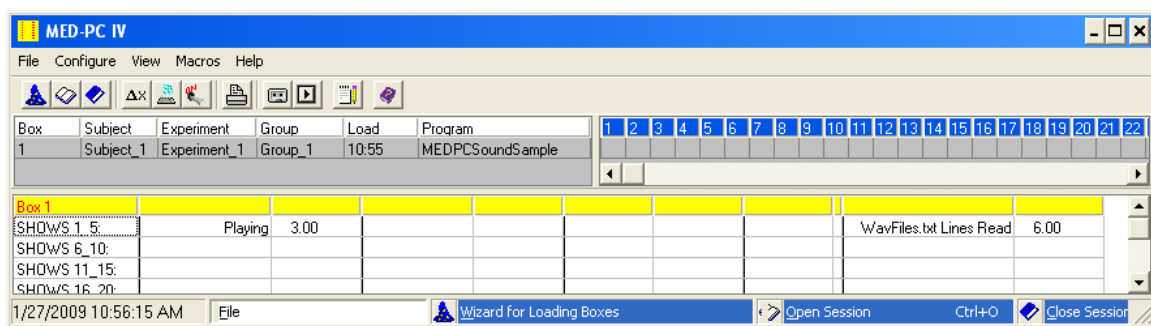


*Figure 3.9 - Send Start Command Screen for Multiple Box Configuration*



In both cases (Figure 3.8 and Figure 3.9), the screens are where the researcher decides to either load more boxes, send a start signal to boxes that are already loaded, or enter the MED-PC run-time environment without sending a start signal by selecting "I am finished with the wizard". This option results in the screen shown in Figure 3.10.

Figure 3.10 - The MED-PC Run-Time Screen

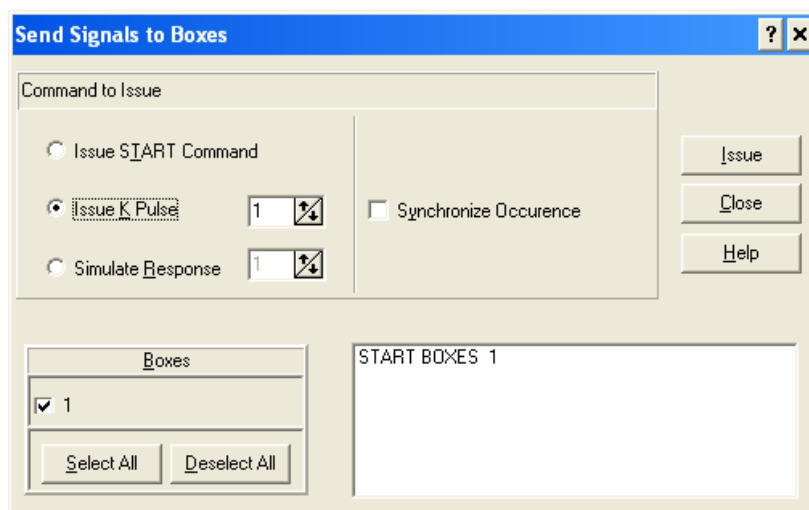


## Playing Wave Files

Issue a **START** command, and the songs in the List File named in the SetWAVFile function will play successively, in repetition, until a K1 pulse is sent to the box running the procedure.

To issue a K1 pulse, pull down the **Configure** menu and select **Signals**. The **Send Signals to Boxes** dialog will appear. Select the running box at the lower left, and the **Issue K Pulse** radio button at the upper left. Set the K Pulse number to **1** and click the **Issue** button to stop the procedure.

Figure 3.11 - Send Signals to Boxes Dialog



## CHAPTER 4

### *Function Reference*

This chapter explains the three functions exported by the Wave File Player library using the example MedState Notation code: **MEDPCSoundSample.MPC**.

#### **SetWAVFile**

The SetWAVFile function tells the Wave File Player library where to find the List File and returns the number of wave files found in the List File. WavFiles.txt contains 6 files, therefore **X** will be assigned the value 6. This is used later in the procedure as a loop control value, so the files can be played successively in a loop until a K1 pulse is issued.

MEDPCSoundSample.MPC uses the **SetWAVFile** function in State 2.

```
S2,
    #START: ~X := SetWAVFile('C:\MED-PC IV\WavFiles.txt');~;
    SHOW 5,WavFiles.txt lines read,X;
    IF X <= 0 [@Exit, @Continue]
        @Exit: ---> S6
        @Continue: ---> S3
```

---

**NOTE:** a command from MedState Notation to external code (like the Wave File Player library) must be surrounded by tildes (~).

---

#### **IsReady**

The IsReady function asks the Wave File Player if it is ready to play a wave file and returns a 0.0 if it is not and a 1.0 if it is.

MEDPCSoundSample.MPC uses **IsReady** in State 3.

```
S3,    #K1: ---> S6
    0.1": ~F := IsReady();~;
    IF F <> 0 [@Go, @Wait]
        @Go: ~R := PlayWave(A);~ ---> S4
        @Wait: SHOW 2,waiting,W/600; ADD W ---> SX
```

State 3 calls **IsReady** every 0.1 seconds until **IsReady** indicates the Wave File Player library is ready to play a sound. The return value of the **IsReady** call is assigned to the variable "F". If the library is ready (**F <> 0.0**), **PlayWave** is called, specifying which wave file to play. **IsReady** returns the value 0.0 if the Wave File Player library is busy playing another wav file. **IsReady** returns the value 1.0 if it is ready to play a wave file.

## PlayWave

The PlayWave function tells the Wave File Player to play a wave file. The variable "A" indicates which wav file to play from the Wav List File. E.g. to play the first file referenced in the Wav List File, use:

**~PlayWave(1);~**

Using the example of the WavFiles.txt file, **~PlayWave(1);~** would play chimes.wav, and **~PlayWave(6);~** would play tada.wav. **~PlayWave(7);~** would cause an error and no sound would be played, as there are only 6 files in WavFiles.txt.

The **PlayWave** function returns the value 1.0 if it attempts to play a sound file. PlayWave returns the value 0.0 if it does not attempt to play a sound file. PlayWave would not attempt to play the sound file in the following cases:

- Wav List File not set (SetWAVFile not called, or SetWAVFile had a problem reading the file)
- It is not ready (IsReady not called, or return value not checked properly)
- Invalid Wav List File index (index less than 1, or greater than the number of files found by SetWAVFile)

E.g. **~R := PlayWave(7)~** would assign 0.0 to **R** because 7 is too high, there are only 6 files in WavFiles.txt.

In State 3, the variable "R" is assigned the return value from PlayWave.

In State 4, **R** is checked to see if the PlayWave command worked.

```
S3,    #K1: ---> S6
        0.1": ~F := IsReady();~;
        IF F <> 0 [@Go, @Wait]
            @Go: ~R := PlayWave(A);~ ---> S4
            @Wait: SHOW 2,waiting,W/600; ADD W ---> SX

S4,    #K1: ---> S6
        0.1": IF R <> 0 [@Worked, @Failed]
            @Worked: SHOW 1,playing,A; CLEAR 2,2; ADD A ---> S5
            @Failed: SHOW 1,ERROR playing,A; CLEAR 2,2; ADD A ---> S5
```



**CHAPTER 5*****MEDPCSOUNDSAMPLE.MPC***

The program is shown below for reference purposes.

```
\ Copyright (C) 2008 MED Associates, All rights reserved.
\
\ MEDPCSoundSample.mpc
\
\ This file uses the MEDPCSound.dll to play .WAV files. It will play the files
\ listed in WavFiles.txt in sequential order until a K1-Pulse is issued.
\
\ Requires:
\ MEDPCSound.dll           : DLL exports SetWAVFile, IsReady, PlayWave functions.
\ MEDPCSound_iface.hed     : Header file to define DLL Interface.
\ {$I MEDPCSOUND_IFACE.HED} : Line inserted into User.pas.
\ WavFiles.txt             : Text file with .WAV files, one per line.
\
\ MEDPCSound.dll functions usage:
\ double SetWAVFile(char *F): Set the location of the file that contains the .WAV files.
\                           : Returns the number of .WAV files found in the list.
\
\ double IsReady()         : Returns 1 if DLL is ready to play another .WAV file.
\                           : Returns 0 if not ready.
\
\ double PlayWave(double I): Tells the DLL to play a specific .WAV file in the list, where
\                           : I is ONE-based index into list stored in the file set by SetWAVFile call.
\                           : Returns 0 if SetWAVFile was not called successfully.
\                           : Returns 0 if I < 1.
\                           : Returns 0 if I > files listed in the file set by SetWAVFile call.
\                           : Resolve by calling SetWAVFile, or passing valid index into .WAV file list.
\
\ List Working Variables Here
\ A = One-based song index into WavFiles.txt
\ F = Ready state of MEDPCSound.dll
\   0 = Not ready
\   1 = Ready to play another .WAV file
\ W = Waiting time counter, increases every 0.1" that the DLL is busy playing
\ R = Return value from PlayWave
\   0 = Didn't try to play
\   1 = Tried to play
\ X = Number of song files found in WavFiles.txt
\
\ K-Pulses Used in this Program
\ K1 = Exit Program
\
\*****
\      MAIN PROGRAM CONTROL
\*****
S.S.1
S1,
  0.001": SET A = 1 ---> S2
S2,
  \ Wait for START signal, then call MEDPCSound.dll's
  \ SetWAVFile function to indicate which text file
  \ has a list of .WAV files. One file per line Ex:
  \
  \ line 1:  C:\MED-PC IV\chimes.wav
  \ line 2:  C:\MED-PC IV\chord.wav
  \ line 3:  C:\MED-PC IV\ding.wav
  \ line 4:  C:\MED-PC IV\ringin.wav
  \ line 5:  C:\MED-PC IV\ringout.wav
  \ line 6:  C:\MED-PC IV\tada.wav
  \
  \ SetWAVFile will return 6, therefore call PlayWave with 1 - 6
  #START: ~X := SetWAVFile('C:\MED-PC IV\WavFiles.txt');~;
  SHOW 5,WavFiles.txt Lines Read,X;
  IF X <= 0 [@Exit, @Continue]
```

```

        @Exit: ---> S6
        @Cont: ---> S3

S3,      \ Exit on K1-Pulse.
        \
        \ If MEDPCSound.dll's IsReady returns 0, try again in 0.1".
        \ otherwise, call PlayWave with ONE-based index into .WAV text
        \ file, go to State 4 to display results of call.
#K1: ---> S6
0.1": ~F := IsReady();~;
      IF F <> 0 [@Go, @Wait]
        @Go: ~R := PlayWave(A);~ ---> S4
        @Wait: SHOW 2,Waiting,W/600; ADD W ---> SX

S4,      \ Exit on K1-Pulse.
        \
        \ Display results of PlayWave call and increase A for next time
        \ MEDPCSound.dll IsReady to play next .WAV file.
#K1: ---> S6
0.1": IF R <> 0 [@Worked, @Failed]
        @Worked: SHOW 1,Playing,A;
                CLEAR 2,2; ADD A ---> S5
        @Failed: SHOW 1,ERROR Playing,A;
                CLEAR 2,2; ADD A ---> S5

S5,      \ Exit on K1-Pulse.
        \
        \ If the file line index A is past the last line of the .WAV text
        \ file X, start back at one, then head back to State 3 to try to
        \ play next .WAV file.
#K1: ---> S6
0.1": IF A > X [@True, @False]
        @True: SET A = 1 ---> S3
        @False: ---> S3

S6,
0.1": ---> STOPKILL

```